



The American Society of
Mechanical Engineers

Reprinted From
Proceedings of the 30th. Intersociety Energy Conversion
Editors: D. Y. Goswami, L. D. Kannberg, T. R. Mancini,
and S. Somasundaram
Book # 10384A - 1995

INTERMEDIATE LEVELS OF AUTONOMY WITHIN THE SSM/PMAD BREADBOARD

Norma R. Dugal-Whitehead

NASA

Marshall Space Flight Center

EL73

MSFC, AL 35812

Phone : (205) 544-3304

FAX : (205) 961-4424

e-mail: norma.whitehead@msfc.nasa.gov

Bryan Walls

NASA

Marshall Space Flight Center

EB72

MSFC, AL 35812

Phone : (205) 544-3311

FAX : (205) 544-5841

e-mail: bryan.walls@msfc.nasa.gov

ABSTRACT

The Space Station Module Power Management and Distribution (SSM/PMAD) breadboard is a test bed for the development of advanced power system control and automation. Software control in the SSM/PMAD breadboard is through co-operating systems, called Autonomous Agents. Agents can be a mixture of algorithmic software and expert systems.

The early SSM/PMAD system was envisioned as being completely autonomous. It soon became apparent, though, that there would always be a need for human intervention, at least as long as a human interacts with the system in any way. In a system designed only for autonomous operation, manual intervention meant taking full control of the whole system, and losing whatever expertise was in the system. Several methods for allowing humans to interact at an appropriate level of control were developed.

This paper examines some of these intermediate modes of autonomy. The least humanly intrusive mode is simple monitoring. The ability to modify future behavior by altering a schedule involves high-level interaction. Modification of operating activities comes next. The coarsest mode of control is individual, unplanned operation of individual Power System components. Each of these levels is integrated into the SSM/PMAD breadboard, with support for the user (such as warnings of the consequences of control decisions) at every level.

INTRODUCTION

The Space Station Module Power Management and Distribution (SSM/PMAD) breadboard is a testbed for the development of advanced power system control and automation techniques. These techniques range from the use of "intelligent" power system hardware to advanced power system software techniques.

The earliest SSM/PMAD system goal was to design a completely autonomous power system in which no human intervention was necessary. Users regularly found the need to make modifications to what the system was doing, however. This required disabling all of the high-level computational capability, such as fault detection, schedule implementation, and embedded procedural knowledge. Clearly, this was not the user's desire. The system has since evolved so that the human becomes a peer among several autonomous agents. The user can now adjust the granularity of control to interact with the system at the varying levels of control.

SSM/PMAD Breadboard

The SSM/PMAD breadboard consists of an electrical power system based on the design of the distribution system for a Space Station Freedom Common Module, with the addition of some advanced switchgear and a very sophisticated control and monitoring system. The Common Module design was the direct precursor of the US Habitation and Laboratory modules being constructed for International Space Station Alpha (ISSA), so the EPS topolo-

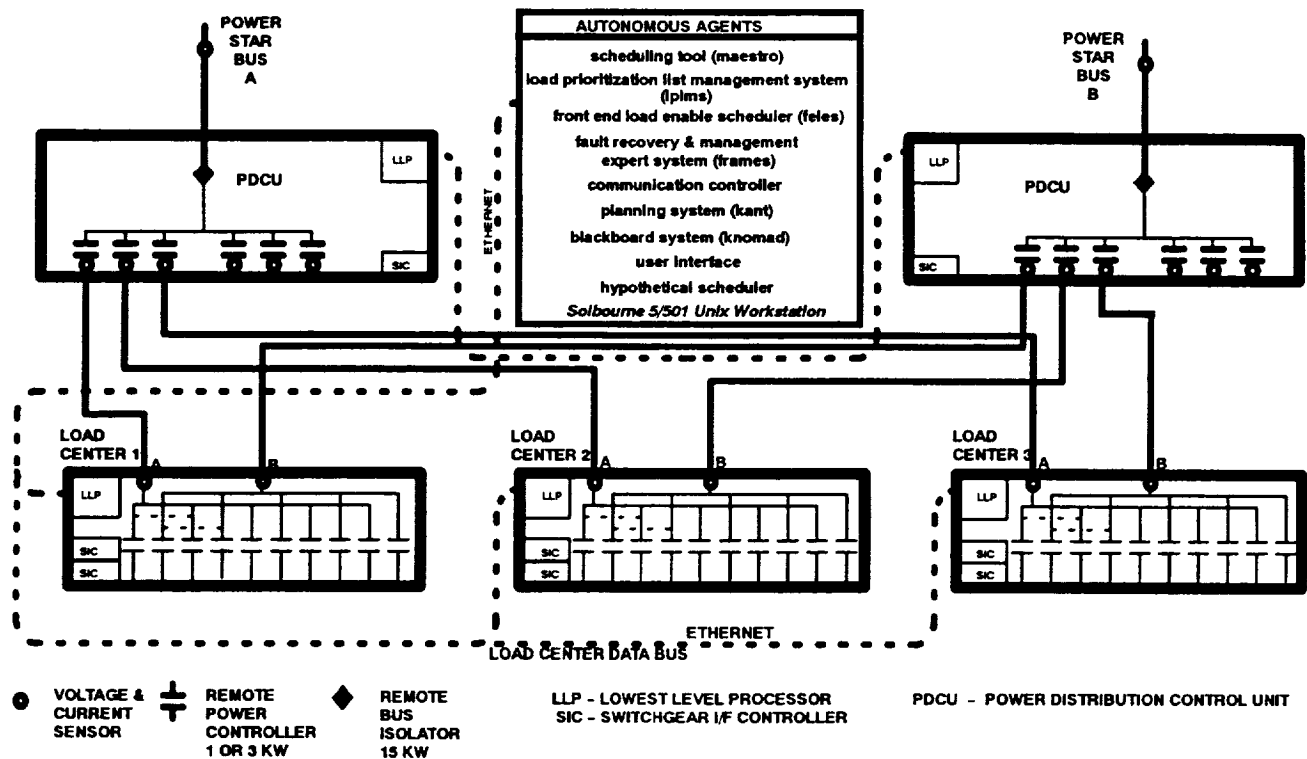


Figure 1 SSM/PMAD Breadboard

gy of the SSM/PMAD breadboard and the ISSA modules share some basic similarities, such as a 120 VDC voltage level and radial distribution.

Figure 1 shows the SSM/PMAD layout. Two buses provide redundant power to each of three load centers. The switchgear consists of three layers of components. At the top level of each bus is a Remote Bus Isolator (RBI) which is a simple, remotely controllable switch. The next two levels consist of Remote Power Controllers (RPCs). RPCs are "intelligent" switchgear in that they sense current and voltage parameters, can act as circuit breakers based on several different conditions, and can remotely communicate the internal data and the reason for any trips. The layer directly below the RBI is made up of RPCs rated at three times the current-carrying capacity of the RPCs making up the bottom level. Components of the SSM/PMAD system not shown in Figure 1 are the power sources, the loads, and the fault insertion system. The sources are two large DC power supplies. A resistive load

bank simulates most of the loads, with a few subsystems simulated by representative hardware. The fault insertion system includes manually operated switches that are wired into various sections of the distribution system to allow the insertion of short circuits at appropriate places, and an enunciator board that shows what switches in the EPS are open or closed.

The software in the SSM/PMAD breadboard consists of a set of co-operating software systems, called Autonomous Agents. The Autonomous Agents consist of the algorithmic software agents and the expert system agents shown in the center box in Figure 1, as well as the software in each LLP (Lowest Level Processor).

Schedule-based Control

A schedule of events, or timeline, is the basis of operation in the SSM/PMAD breadboard. Three of the Autonomous Agents in the breadboard directly deal with schedules: the Maestro scheduling tool, the Front-End

Load Enable Scheduler (FELES), and the Hypothetical Scheduler. Maestro can be used interactively to build schedules. It also acts independently in the breadboard to generate a new schedule when there are changes in available resources that make the running schedule infeasible. FELES takes information from the current schedule to build a short-term list of switch positions and expected current draw for each piece of switchgear in the system, sorted by Load Center or Power Distribution Control Unit. It then transfers a list to each Lowest Level Processor (LLP) for local implementation. The Hypothetical Scheduler offers a subset of the tools available in Maestro for building schedules, but is intended for building a schedule to "graft in" to the current schedule.

A user begins building a schedule by creating activities. An activity defines the process by which something is done, and the resources required. For example, placing a piece of equipment in a rack, conducting a particular experiment, or maintaining the temperature in a room are all activities, each requiring some set of resources. Creating an activity involves specifying a process and designating the resources needed for each process element, or subtask. Any timing constraints between the various subtasks can be specified, also. The Maestro tool is quite flexible in allowing requirements and interrelations to be described either specifically, or as a range. Each activity must also be assigned a priority.

Various kinds of resources can be specified for a subtask, and limits can be set on the availability of the resource. For instance, a subtask of an experiment might require the availability of two crew members, or the use of a certain tool. The user can define new classes of resources and define the availability. Since our focus here is electrical power, a special class of resources, Powered Equipment, describes all electrical loads. Each piece of Powered Equipment can have different modes of operation which may vary in power consumption and priority. When specifying a piece of Powered Equipment in a subtask, the point of connection (that is, which RPC it is connected below) must also be specified. This allows the system to track power as

a resource. The schedule takes into account power system topology to make sure power usage is also within allowable limits, and no lines or breakers are overloaded. Once a resource is defined, it can be stored in a database for further use in other subtask descriptions. Activities themselves can be saved for later use or modification.

The user submits a list of candidate activities for scheduling to Maestro. Maestro creates a feasible schedule, heuristically using priority information to come to as good a solution as possible. If the schedule is acceptable, the user can save it for later use. A valid schedule is required for normal operation of the breadboard.

LEVELS OF AUTONOMY

This paper examines the varying levels of autonomy within the SSM/PMAD breadboard. The least humanly intrusive mode of operation is simple monitoring of the system values through the user interface. The ability to modify future behavior by altering a schedule involves high-level interaction. The coarsest mode of control is unplanned operation of individual power system components. Each of these levels is integrated into the SSM/PMAD breadboard, with support for the user at every level. The user can choose the level of interaction, and can change from one mode to another.

Monitoring

As long as the system is operating within scheduled parameters, with events both internal and external to the breadboard conforming to the projected timeline, the user need only monitor the system. The user has the option of monitoring the power system's voltages and currents on the "Power System" screen, viewing the current schedule on the "FELES" screen or monitoring the power usage of the system on the "Power Utilization" screen. The "Power System" screen is a depiction of the present SSM/PMAD breadboard configuration; power system values are displayed both numerically and graphically on this screen. On the "FELES" screen, the schedule is displayed as a set of activity timelines with the present mission time designated by a moving line marked "Now!" which updates every minute. The power consumption levels

for the buses, the load centers and the individual RPCs are available on the "Power Utilization" screen. On this screen, scheduled versus actual power consumption levels can be represented on the same graph.

Monitoring is the primary mode of operation of the SSM/PMAD breadboard since the breadboard contains autonomous agents capable of fault diagnosis, isolation and recovery (FDIR). Monitoring the system doesn't involve a significant component of control for the human user. Control is available to the user at several levels, however.

Modifying the Schedule

Scheduling takes place at a high conceptual level of control. Since there is a database of activities available, a schedule can be built up just thinking in terms of what needs to be done. New activities can be built up by modifying existing descriptions, and using preexisting subtasks. Since a particular piece of equipment will normally be attached to the power system at a particular location, the user doesn't have to think about power system topology or remember the details of particular power modes to schedule it.

For those times when a change of schedule is necessary the operator can use the Hypothetical Scheduling tool. This tool has the capabilities of Maestro, but can be used while the original schedule continues to run. First, the operator has to choose when the new schedule will be grafted in. This is an important choice; if the chosen time is too close to the present, the operator may not have time to make the changes and verify them before the time passes.

The simplest form of schedule modification is to replace the current schedule with another schedule from the schedule database. Next in simplicity is to change the start time for activities, or to add or remove activities from the schedule. The Hypothetical Scheduler will tell the user what positions on the modified schedule are possible, and what will have to be moved or deleted to allow the proposed changes. The tool won't allow an impossible schedule to be created. Once a valid schedule has been created, the user can

choose whether or not to implement the modified version.

The user can go beyond simply adding, changing or deleting activities that already exist. New activities can be created, or old activities modified. It is even possible to modify an activity that is currently running, though timing constraints become more troublesome.

Seizing of Switches

Working with the schedule, the user is effecting power system control without having to think about any details of the power system. The user can also choose to work directly with power system components by manually controlling switches (RPCs or RBIs) or groups of switches. To notify the system that such control is desired, the switch or switches must be "seized."

Grouping of Switches From the power system interface window, the command, "Group Switches" allows one or more switches to be grouped together and assigned a name. If the user chooses to seize control of a group, the group acts as a unit. If one of the switches in a group goes out of service, the whole group is removed from manual control, and the resources reserved for the group become available for rescheduling. Seizing a group is similar to building an activity with a single subtask, where all the resources are powered equipment, except that one is dealing directly with power system components rather than with the equipment. Seizing a group of switches is an intermediate level of control between schedule manipulation and seizing control of individual switches.

Seizing Manual Control By selecting the "Seize Manual Control" button on the side panel of the power system screen, the operator can choose the switches and groups of switches which the operator needs to seize. When the selections have been accepted a configuration workbox appears. This workbox allows the user to specify: the length of time each switch is required, at what power level, at what static priority, at what relative priority, whether or not this switch has a redundant switch to power the load if this switch is tripped off, and whether or not the system is allowed to test this switch if there is a

fault. After this information is accepted the system prompts the user to run "Calculate." The planning system, called Kant, evaluates the effects of the seizure of these switches, and presents the results.

Kant responds to the proposed seizure in one of several ways. It may be that there is no impact to the schedule if the switch was not in use, was not planned for use during the interval chosen, and adequate power is available. The response may be that an activity that uses this switch or a switch in this group will be interrupted, if the switch is currently being used or is scheduled to come on during the time the switch is to be under manual control. Or, there may be a list of activities which will be affected if the amount of power required exceeds the amount which is available, and the priority of this manual seizure is high enough to bump the existing experiments from the schedule. There is also the possibility that permission for the seizure of the chosen switch will be denied because the priority level for this seizure is too low or the priority of the activity using this switch is too high or the priorities of the other activities in this load center are too high.

If the affects of seizing the proposed RPCs are not acceptable, the operator has the choice of aborting the seizure or changing the parameters associated with any or all of the switches which are to be seized and recalculating. Also during this process, switches may be added and deleted from the list, as needed.

If the affects have been deemed acceptable and the seizure confirmed, once the time period for the seizure has arrived the RPC is marked for manual control. At this time, the controls on the side panel for manual control of an RPC are enabled, and the RPC can be turned off and on, as needed. The operation of a group of switches while under manual control is just the same as a set of single switches. Individual switches in the group can be operated separately.

Update Manual Control Another very useful control feature which the operator may use in controlling RPCs manually is the ability to "Update Manual Control." Updating manual control allows the operator to

change the time period of use of the seized switches, change the power level or the priority level of the switch, and calculate the effects of these actions. As with all of the operations which have been described in this paper, the user has the ability to implement or cancel the proposed changes after the affects on the system are known. The system also has the ability to deny the implementation of the changes if the requested action will impact the running of a higher priority activity, just as in the original seizure of the switch.

When an RPC trips and the switch is diagnosed as being faulty or a fault is diagnosed below the RPC, the RPC is automatically placed in the manual control mode with a power consumption level of 1 W. The ability to update manual control can be very useful when it is time to test this faulted switch. A faulted switch can not be seized but the schedule for a switch under manual control can be updated to allow for the amount of current this switch can draw to be increased enough for the switch to be tested with a load. In this manner the system may be tested and fixed. Of course, when the attempt to update the operating conditions of this switch is made the results of the calculation may be any of those responses discussed above.

Release Manual Control The operator can, also "Release Manual Control" of a switch or group of switches from the power system screen. This feature allows the operator to relinquish the control of a switch or group of switches before the established time period is over. This might be used if the needed experiments have been completed ahead of schedule. This act may not trigger Maestro to recalculate the schedule since the present schedule is still valid.

The manual control of RPCs can be used for various reasons. The most obvious reason is to try to troubleshoot a problem in the RPC or the load. There is the possibility that an experiment may need to be run in a manner where power is only delivered on command and not continuously. The manual seizure of a switch can also be used to open a switch above a faulty load or a resistive short that is below the tolerance level the hardware or software can detect. Seizure of an RPC for this reason could be used as an immediate re-

sponse to the problem, until the schedule could be modified using the Hypothetical Scheduler.

When manual control is used to troubleshoot a tripped RPC, the use of the "Release Manual Control" button will not return this switch to operation. There is a utility called "Returning Switches to Service," which must be run to allow repaired switches to be returned to the schedule. When "Return Switches to Service" is run, the RPC is turned on, if testing is allowed, for a short time to make sure that it does not trip again and then, that RPC is returned to service. Maestro is informed of the availability of the new switch and the schedule is recalculated.

CONCLUSIONS

There is no way any system can know what a human wants without communicating with the user. Including a human as part of a computer-based system is difficult, though, because humans are slow and error prone. And, although humans have powers of analysis no computer has ever approached, they quickly get bogged down in details and overwhelmed with too much data or too much repetition.

Intermediate modes of autonomy provide a solution for this problem. The user is free to choose a level of interaction based on the user's goals and desires. In the fully autonomous mode, the system operates without human intervention, though the system is capable of requesting assistance when it is needed. A succession of modes allowing more detailed responsibility and control are available for the user down to the lowest level of control that might be desired, while maintaining the knowledge and expertise of the embedded systems. The system constantly provides the operator with the information and feedback appropriate to the user's present level of control and provides the tools necessary for the user to evaluate potential actions before they are implemented.

This concept has been demonstrated in the SSM/PMAD breadboard. The human is considered to be one of the interacting autonomous agents, with special support from the user interface and from the Kant planning system. The human may just monitor the sys-

tem, or may control the system at the conceptual level of schedules, activities, subtasks, pseudo-tasks (manually seized groups), or individual components. The human's interactions are automatically integrated into the running schedule (the highest level) to be implemented by the proper components (the lowest level) without the human having to be aware of actions taking place at other conceptual levels.

This implementation of intermediate modes of autonomy was the result of incremental development. Though it proves the value of the concept, the flow between the various modes is not as smooth as it might have been with a top-down design. For instance, the interface to modify the subtask of an activity is radically different from that for modifying a manually seized group, though the two actions are adjacent conceptually. Nonetheless, this implementation clearly shows that the use of intermediate modes of autonomy is an important part of effectively monitoring and controlling a complex system. Further research into the concept is warranted.

REFERENCES

Ashworth, B; Riedesel, J.; Myers, C.; Jakstas, L.; Smith, D.; *Space Station Automation of Common Module Power Management and Distribution, Interim Final Report*, Martin Marietta MCR-89-516 under NASA Contract NAS8-36433, July 1990.

Miller, B; Britt, D.; Elges, M.; Myers, C.; *Space Station Module Power Management and Distribution System (SSM/PMAD), Final Report*, Martin Marietta MCR-94-1315 under NASA Contract NAS8-36433, May 1994.

